

## CS 200 – Intro to Programming

---

### *Assignment 3 Fall 2018*

*Due Date: Thursday 8th November 2018, 11:55 PM*

*Please keep in mind the following guidelines:*

- *Do not share your program code with anyone.*
- *Do not copy code from the internet.*
- *If you receive any assistance, mention the part of code in which you received assistance.*
- *You must be able to explain any part of your submitted code.*
- *All submissions are subject to automated plagiarism detection.*

### *Submission:*

*You have to submit all the .cpp files containing source code. Zip all .cpp files into one file named as <your8DigitRollNumber>\_Assignment1\_CS200.zip and submit the zip file.*

*Total Marks: 100*

### Question 1: [20 marks]

You are required to make a class named Food that contains 3 functions Fruits(), Vegetables() and Nuts().

These functions must have the capability to initialize an array of string (that will contain names) and array of integer (that will contain calories) which must contain the following things (format “name\_calories”).

- Apple\_25, guava\_55 , banana\_10, mango\_60, peach\_15 should be placed in Fruits().
- Broccoli\_44, onion\_14, peas\_47 , potato\_88, capsicum\_20 should be placed in vegetables().
- Almonds\_80, cacao\_85, coconut\_62, hazelnut\_77, cashew\_90 should be placed in Nuts().

Once the arrays are initialized, each function should display the data.

Now you have to create a child class inherited from “Food”, called “Detail” with member functions named Take() and Print(). The data must be initialized using the child class constructor. The take() function must first ask the user what amount of calories he intends to consume. Now the print function must all the combinations possible that is less than or equal to the desired calorie intake of user. Combinations must contain one fruit, one vegetable, and two nuts, with their total calories too. The print function must also show the total number of combinations made too. Note that the base class variables must only be accessible by the derived class.

## Question 2: [30 marks]

You have to implement the multiple and multilevel inheritance using the details given below.

### HUMAN class

- Must be an abstract class
- Must contain a pure virtual function called print()
- Must contain following protected variables
  - String name
  - Int age
- Must contain parameterized constructor for all variable initialization.
- Must contain get and set functions for each variable.

### STUDENT class

- Must contain print() function.
- Must contain following protected variables
  - Int rollnum
  - String unname
  - String array of courses.
  - Int stipend
- Must contain parameterized constructor for all variable initialization.
- Must contain get and set functions for each variable.

### EMPLOYEE Class

- Must contain print() function.
- Must contain following protected variables
  - Int empID
  - String compName
  - String array of Languages.
  - Int salary
- Must contain parameterized constructor for all variable initialization.
- Must contain get and set functions for each variable.

### Businessman Class

- Must contain print() function.
- Must contain following protected variables
  - Int regID

- String compName
- Int Earning.
- Must contain parameterized constructor for all variable initialization.
- Must contain get and set functions for each variable.

### PERSON Class

- Must contain print() function.
- Must contain following protected variables
  - Bool is\_student
  - Bool is\_employee
  - Bool is\_businessman
- Must contain parameterized constructor for all variable initialization.
- Must contain get and set functions for each variable.

### Following things must be kept into consideration

- ➔ HUMAN Class is the top-most parent class.
- ➔ STUDENT, EMPLOYEE, and BUSINESSMAN classes have protected inheritance from HUMAN Class.
- ➔ PERSON class is inherited from STUDENT, EMPLOYEE, and BUSINESSMAN classes. (multiple inheritance.). The inheritance should be private type.
- ➔ All the initialization must be done using constructors. The variable of the same class must be initialized from constructor by calling its setter functions explicitly.
- ➔ The user should be asked whether the person is a student, an employee, a businessman or all, and initialize the variables accordingly through parameterized constructor.
- ➔ The user must also have an option to change the credentials after their initialization through set functions.
- ➔ The print function of person must call all print functions of its parent classes within its own print function. To display its own members, the print function must utilize the get functions.
- ➔ The print function should show complete details of that particular person with an extra detail of “Total tax payable by that person”. The tax must be calculated in the following way:

- If the person is student then tax rate is 2% if stipend is above 50000.
  - If the person is employee then tax rate is 5% if pay is above 50000 and below 100000, and 10% above 100000.
  - If the person is businessman then tax rate is 8% if pay is above 50000 and below 100000, and 12% above 100000.
  - Add the total tax and display as Total Tax Payable.
- ➔ Create this data for at least 5 persons (this should be asked at runtime). So it means you have to make an object array of PERSON class.
- ➔ Create a template function that takes template array of these 5 persons and prints the data in ascending or descending order according to its total tax payable. The return type must be Void.

### Question 3: [50 marks]

This project is an exercise in classes and inheritance. It should be implemented using multiple (specification, implementation, and application) source code files. All data members should be private.

The goal of the program is to print pay checks for a company. To accomplish this task, you will need to create and derive several classes.

**1.** file names: Employee.h, Employee.cpp Design a base class named Employee. The class should keep the following information in its data members:

- Employee name
- Social Security Number, in the format xxx-xx-xxxx, where x is a digit within the range 0 through 9
- Employee number, in the format xxx-L, where x is a digit within the range 0 through 9, and the L is a letter within the range A through M. Add a constructor and other appropriate member functions to manipulate the class data members.

Input Validation: Only accept valid Social Security Numbers (with no alphabetic characters) and valid Employee Numbers (as described above). Note: Input

validation should be done in the setter functions and no input statements should be used.

**2.** file names: EmployeePay.h, EmployeePay.cpp Design a class named EmployeePay. This class should be derived from the Employee class. It should store the following information in data members:

- Annual pay
- Weekly pay (To be calculated from Annual Pay) Add a constructor and other appropriate member functions to manipulate the class data members. Input Validation: Do not accept negative values for annual pay.

**3.** file names: HourlyPay.h, HourlyPay.cpp Design a class called HourlyPay. This class should be derived from the EmployeePay class. It should store the following information in data members:

- Hourly pay rate (based on Weekly pay, assume a 40-hour work week)
- Overtime pay rate (1.5 times hourly rate if over 40 hours)
- Number of hours worked Add a constructor and other appropriate member functions to manipulate the class data members. Input Validation: Do not accept values over 60 for hours worked.

**4.** file name: EmployeeMain.cpp Finally, using the classes you have created, write a program which uses a function with the following prototype to print a pay check: void PrintCheck(HourlyPay); The program should ask for sample data for the employee information and then calculate the employee's pay and display her/his pay check on the screen. Be creative!!